

APPLICATION  
FOR  
UNITED STATES LETTERS PATENT

TITLE: WEB SERVICE INTEGRATION

APPLICANT: PETER SCHWARZE

CERTIFICATE OF MAILING BY EXPRESS MAIL

Express Mail Label No. EV 398 159 260 US

November 12, 2003  
Date of Deposit

## **WEB SERVICE INTEGRATION**

### **CROSS REFERENCE TO RELATED APPLICATIONS**

[0001] This application claims the benefit of priority from U.S. Provisional Application entitled "Web Service Integration", filed November 18, 2002, Application Serial No. 60/427,509, which is incorporated herein by reference.

### **BACKGROUND**

[0002] Computers may use browsers to interface with the World Wide Web and other Internet-based networks, such as enterprise portals. Many browsers allow users to add links to their favorite or most used web pages in a "Favorites" list. Links can also be imported into a browser's "Favorites" list from a different computer or browser.

[0003] In an organization, "Favorites" lists may be very useful for personalizing user computers. The individual users may select those links they use most often in their work, which may improve work efficiency. However, since the user must add or import the links at the client computer, the "Favorites" list feature in many browsers may not be an efficient way to make links available across the organization.

### **SUMMARY**

[0004] An enterprise system may include a server including an integration module that enables applications at the server to integrate web services, e.g., by providing links to the external web service. Each application is associated with one or more business types in a table at the server. The integration module, which may be an API, sends a list of external web services of a type (or types) associated with an application when a user at a client computer in the enterprise system opens that application, e.g., through an enterprise portal. The integration module may also generate a URL call and redirect the user's browser to an external web service when the user selects that external web service.

[0005] The URL call may be in an OCI/OPI-compliant format. The integration module may also receive a response from the selected external web service in an OCI/OPI-compliant format.

### **BRIEF DESCRIPTION OF THE DRAWINGS**

[0006] Figure 1 is a block diagram of a system including integrated external web services.

[0007] Figure 2 is a flowchart describing a technique for providing access to integrated external web services.

[0008] Figure 3 is a block diagram illustrating an operation for redirecting a browser to an integrated external web service in different web platforms.

[0009] Figure 4 is a block diagram for importing data from an integrated web service in different web platforms.

[0010] Figure 5A shows an exemplary Open Catalog Interface (OCI) inbound interface

[0011] Figure 5B shows an exemplary OCI outbound interface.

#### DETAILED DESCRIPTION

[0012] Figure 1 shows a computer-based system 100 including an enterprise system 102. The enterprise system 102 includes a server 104 that hosts applications 106 that may be utilized by users (e.g., employees of a company) at client computers 108. The client computers may include browsers 110 that enable the clients to communicate with the server via, e.g., a web portal 112.

[0013] The server 104 may include an API (Application Programming Interface) 114 that enables the applications 106 at the server to integrate web services 116, e.g., by providing links to the external web service. This web service integration feature may provide a functionality similar to a personal "favorites" list of links. However, rather than individual users having to select individual links or import links into their browsers, the links are provided automatically

through the application. This may provide a convenient way to make certain "favorite" links available across the organization.

[0014] Information about a number of external web services may be stored in a table 120 at the server 104. Every field in the table 120 may contain a name-value pair and have a type. The information stored in the table for each external web service may include the following information: the external web service URL (Uniform Resource Locator), which should refer to the location of the external web service; fields specific to the external web service, such as username and password; and a business type.

[0015] The business type indicates a classification for the web service based on the type of information the web service provides. The following are exemplary classifications that may be used for external web services: catalogs; business directories; search engines; chat rooms; newsgroups; marketplaces; portals; companies; associations; financial services; e-forms; etc. These classifications may be selected and assigned to selected external web services during a customization process for the web service integration feature.

[0016] Figure 2 is a flowchart describing a technique for providing access to integrated web services according to an embodiment. When a user opens an application (block 202), the application may provide business type(s) as a parameter to the

API (block 204). The API may then return a list of external web services assigned the business type(s) and information associated with those web services (block 206). This information may be displayed for the user in the form of hyperlinks and/or descriptions of the web services.

[0017] The user may trigger a call to an external web service by selecting a link to the web service (block 208). The API may return the URL of the external web service and all parameters from the table 120 that will be transferred to the web service (block 210). The server 104 may then construct a URL call to the external web service from the URL and parameters (block 212) and redirect the client browser to the external web service (block 214). The application may open a new browser at the client for the external web service (block 216).

[0018] In an embodiment, the server may include SRM (Supplier Relationship Management) software developed by SAP AG of Walldorf, Germany. The enterprise system 102 may utilize an interface between the SRM server and the client computers, as shown in Figure 3. The interface may be an SAP web platform, e.g., an ITS (Internet Transaction Server) 302, a WAS (Web Application Server) 304, or a Java-based platform, such as Webdynpro. For the Webdynpro platform, the functionality of the API may be provided as a java package at the Java server 306.

[0019] The enterprise system 102 and the external web services 116 may communicate using an interface protocol, such as the Open Catalog Interface (OCI)/Open Partner Interface (OPI) developed by SAP AG of Walldorf, Germany. In an embodiment, the API 114 will also allow the import of data from external web services that are OCI/OPI compliant, as shown in Figure 4.

[0020] OCI/OPI uses standard Internet protocols, e.g., HTTP (Hypertext Transfer Protocol), to exchange information between the server and the external web services (e.g., catalog servers and external partner directories). Using the OCI/OPI, the enterprise system may send a request in an OCI/OPI-compliant format to an external web service, and the external web service may return a response page, including results compiled in response to the request, in an OCI/OPI-compliant format.

[0021] The OCI/OPI includes an outbound section and an inbound section. The outbound section consists of the information in the table 120 at the server 104 (Figure 1). As described above, the enterprise system uses this information to construct a URL call to the external web service and then redirect the client browser to this URL. In an embodiment, the external web service may be accessed using the HTTP methods GET or POST, which includes the outbound interface field data. The OCI/OPI-compliant external web service then parses and decodes

this data and may compile information to return to the enterprise system.

[0022] The inbound section consists of information that is sent to the enterprise system by the external web service. The inbound section may be sent back to the electronic procurement system in an OCI/OPI-compliant, e.g., an HTML page or an XML file. For example, the external web service may be a supplier that provides an electronic catalog. For each item selected in the catalog by the user and sent to the enterprise system, all required fields must be sent, along with the optional fields. The fields may include the following information: a description of the item; a quantity to add; a unit of measure of the item; a price; a currency; a product master number in the electronic procurement system; and a number of days until the item will be available. Figure 5A shows exemplary fields for an OCI inbound interface, and Figure 5B shows an exemplary OCI outbound interface, in this case an HTML response page.

[0023] As described above, the external web service integration feature may provide a convenient way to make certain "favorite" links available across the organization. Furthermore, the business type parameter provides a way to tailor the links made available to the particular application being used. The business type parameter also provides a way to update the links provided by a particular application. For example, for an application that provides links with the



"search engine" business type, the links provided may be updated by adding and/or removing links with the "search engine" business type to or from the table.

[0024] The computer programs (also known as programs, software, software applications or code) described above may include machine instructions for a programmable processor, and can be implemented in a high-level procedural and/or object-oriented programming language, and/or in assembly/machine language. As used herein, the term "machine-readable medium" refers to any computer program product, apparatus and/or device (e.g., magnetic discs, optical disks, memory, Programmable Logic Devices (PLDs)) used to provide machine instructions and/or data to a programmable processor, including a machine-readable medium that receives machine instructions as a machine-readable signal. The term "machine-readable signal" refers to any signal used to provide machine instructions and/or data to a programmable processor.

[0025] The systems and techniques described here can be implemented in a computing system that includes a back-end component (e.g., as a data server), or that includes a middleware component (e.g., an application server), or that includes a front-end component (e.g., a client computer having a graphical user interface or a Web browser through which a user can interact with an implementation of the systems and techniques described here), or any combination of such

back-end, middleware, or front-end components. The components of the system can be interconnected by any form or medium of digital data communication (e.g., a communication network). Examples of communication networks include a local area network ("LAN"), a wide area network ("WAN"), and the Internet.

[0026] The computing system can include clients and servers.

A client and server are generally remote from each other and typically interact through a communication network. The relationship of client and server arises by virtue of computer programs running on the respective computers and having a client-server relationship to each other.

[0027] Although only a few embodiments have been described in detail above, other modifications are possible. The logic flow depicted in Figure 2 does not require the particular order shown, or sequential order, to achieve desirable results. In certain implementations, multitasking and parallel processing may be preferable.

[0028] Other embodiments may be within the scope of the following claims.